

```

/*
    GOPIKRISHNA V
    S3 CSE A
    52
*/
#include <stdio.h>
#include <stdlib.h>
#define max 10
int arr[max];
int isempty = 1;
void insert()
{
    int item, key, i;
    printf("Enter the element = ");
    scanf("%d", &item);
    key = item % 10;
    if (arr[key] == -1)
    {
        arr[key] = item;
        isempty = 0;
    }
    else
    {
        for (i = key + 1; i < max; i++)
        {
            if (arr[i] == -1)
            {
                arr[i] = item;
                isempty = 0;
                break;
            }
        }
    }
}

```

```

    }
}
void EmptyTable()
{
    int i;
    for (i = 0; i < max; i++)
    {
        arr[i] = -1;
    }
}
void display()
{
    int i;
    if (isempty)
    {
        printf("Table Empty\n");
    }
    else
    {
        for (i = 0; i < max; i++)
        {
            if (arr[i] != -1)
            {
                printf("[%d] ", arr[i]);
            }
            else
            {
                printf("[NULL] ");
            }
        }
    }
}

```

```

        printf("\n");
    }
void search()
{
    int key, i, found = 0;
    printf("Enter the element to be seached = ");
    scanf("%d", &key);
    for (i = 0; i < max; i++)
    {
        if (key == arr[i])
        {
            found = 1;
            printf("Element found at %d th position\n", i+1);
            break;
        }
    }
    if (found == 0)
    {
        printf("Element not found\n");
    }
}
void main()
{
    int ch;
    EmptyTable();
    start:
    printf(" \n### MENU ###");
    printf("\n1.Insert");
    printf("\n2.Display");
    printf("\n3.Search");
    printf("\n4.Exit\n");

```

```

printf("Choice >>> ");

scanf("%d", &ch);

switch(ch)
{
    case 1:insert();

        break;

    case 2:display();

        break;

    case 3:search();

        break;

    case 4:exit(0);

        break;

    default:printf("Wrong Input\n");

}

goto start;

}

```

## OUTPUT

```

administrator@administrator-HCL-Desktop:~/gopikrishna$ gedit hashing.c
administrator@administrator-HCL-Desktop:~/gopikrishna$ gcc hashing.c
administrator@administrator-HCL-Desktop:~/gopikrishna$ ./a.out

### MENU ###
1.Insert
2.Display
3.Search
4.Exit
Choice >>> 1
Enter the element = 3

### MENU ###
1.Insert
2.Display
3.Search
4.Exit
Choice >>> 1
Enter the element = 10

### MENU ###
1.Insert
2.Display
3.Search
4.Exit
Choice >>> 1
Enter the element = 15

### MENU ###
1.Insert
2.Display
3.Search
4.Exit
Choice >>> 1
Enter the element = 3

Enter the element = 3

### MENU ###
1.Insert
2.Display
3.Search
4.Exit
Choice >>> 1
Enter the element = 6

### MENU ###
1.Insert
2.Display
3.Search
4.Exit
Choice >>> 1
Enter the element = 9

### MENU ###
1.Insert
2.Display
3.Search
4.Exit
Choice >>> 2
[10] [NULL] [NULL] [3] [NULL] [5] [15] [6] [NULL] [9]

### MENU ###
1.Insert
2.Display
3.Search
4.Exit
Choice >>> 3
Enter the element to be seached = 3
Element found at 4 th position

### MENU ###
1.Insert
2.Display
3.Search
4.Exit
Choice >>> 4
administrator@administrator-HCL-Desktop:~/gopikrishna$

```